

松 山 大 学 論 集
第 26 卷 第 3 号 抜 刷
2 0 1 4 年 8 月 発 行

集団的な囚人のジレンマゲームにおける 戦略推移について

——「報恩」と「報復」を軸として——

安 田 俊 一

集団的な囚人のジレンマゲームにおける 戦略推移について

——「報恩」と「報復」を軸として——

安 田 俊 一

1 は じ め に

本稿は筆者が進化経済学会で行った2回の報告¹⁾を元にあらたにシミュレーションを行い、拡充・発展させたものである。

この研究における一連のシミュレーションの契機となった Nowak and Sigmund (1992) のモデルは、囚人のジレンマゲームを行う個体行動を支配する表現型をペアとなった確率で表現している。これらの確率はともに「対戦相手の1回前の行動」に対して「自分が今回“協力”を選択する確率」を示すため、その大きさによって個体の性格を、また、その分布によって集団の性質を示すことができる。

彼らは集団変化を population dynamics でとらえているので、各個体（ゲームのプレイヤー）の戦略は初期に設定されたままで固定されている。そのため、戦略自体の変化を直接にとらえているわけではない。この点で、彼らの分析フレームワークを基礎として集団進化に GA (Genetic Algorithms) を持ち込んだ分析を行ったのが筆者のこれまでの一連のシミュレーションである。

1) 第9回東京工業大学大会報告「どのような集団が“協力”を生み出すか？－GAによる囚人のジレンマ実験－」(安田俊一 (2005), 2005年3月), 第10回北海道大学大会「協力への到達過程－GAによる囚人のジレンマ実験補論－」(ポスター, 2006年3月)。なお、この一連の研究に関しては2005年度の松山大学特別研究助成をうけた。本論文もその成果の一部である。

本稿では、個体の性格を表す軸に「報復」「報恩」との意味づけをあらたに与えて、初期集団からの戦略推移を集団内の個体分布として観察する²⁾。焦点を当てる戦略推移は、集団が「協力」へ収束し、それが崩壊する状態、および2集団の対戦における、片方の集団による他方の集団の「搾取」の状態である³⁾。また、これまでのシミュレーションと異なり、遺伝子型を2つ持つタイプの個体からなる集団のシミュレーション、および単純2進数による遺伝子型のコーディングとグレイコーディングによるものとの比較を補論として加えた。

2 分析のフレームワーク

Nowak and Sigmund (1992) は、囚人のジレンマゲームを行う個体の戦略を「1回前の相手の行動」で場合分けなされた確率の組 (p, q) で記述する。

p : 前回相手が“協力”した (C) 場合に次に自分が“協力”する確率

q : 前回相手が“裏切った” (D) 場合に次に自分が“協力”する確率

無限繰り返しゲームで戦略 (p_i, q_i) を持つ個体 E^i と戦略 (p_j, q_j) を持つ個体 E^j が対戦した場合の E^i の収束期待利得 $A^i(E^i, E^j)$ は

$$A^i(E^i, E^j) = 1 + 4c_j - c_i - c_i c_j, \quad c_i = \frac{q_i + (p_i - q_i)q_j}{\{1 - (p_i - q_i)(p_j - q_j)\}} \quad (1)$$

となる⁴⁾。

プレイヤーの戦略が (p, q) , $(0 \leq p, q \leq 1)$ の実数の組で表現されているため、 (p, q) 平面の長さ1の正方形内に点をとれば、それを“個体”と見なすことができる。

2) 今回のシミュレーションからプログラム言語を C++ から Java へ変更した。また、集団の状態を表す戦略推移と集団の平均値を同時に観察するためのツールを開発し、その結果世代進行に伴う戦略推移をグラフィカルに表示することを可能にした。本論文で使ったグラフ類はすべてこのツールによって作成した。補論 B 参照。

3) 「搾取」の定義については後述する。

(p, q) の定義からすれば, p 軸 (横軸) は, “前回の対戦相手の協力に対する今回の「返礼」” を意味するため, この軸の性質を「報恩」としよう。 p の大きさは「報恩の程度」を示しており, p が 1 に近ければ, 報恩の程度は高く, 「礼儀正しい (polite)」⁵⁾。逆に p が 0 に近づくほど報恩の程度が低く, いわば「無礼者」に近い態度となる。

q 軸 (縦軸) は “前回の対戦相手の裏切りに対する「仕返し」” を意味するため, この軸の性質を「報復」とする。 q は “協力” する確率であるので, q の値が 1 に近ければ, 裏切りに対して「寛容 (tolerant)」⁵⁾ であるし, 0 に近ければより「報復的」である。

平面における横軸を「報恩」軸, 縦軸を「報復」軸とすると, 正方形内の 1 点で示される個体の性質を分類することができる。それぞれの軸の中点 ($p = 1/2, q = 1/2$) を, 報恩の程度も報復の程度も平均的であるという意味で “平均的な個体” だと考えれば, 平面の端に位置する個体ほど態度が極端になっていく。安田俊一 (2001, 2005) はこの点に着目して, 協力を達成する集団の性質を考察したものがある。そこでのシミュレーションのフレームワークは以下の通りである⁶⁾。

4) 囚人のジレンマの利得行列

	協力 C	裏切り D
協力 C	(R, R)	(S, T)
裏切り D	(T, S)	(P, P)

を $R=3, T=5, S=0, P=1$ とした場合。また, ここで $c_{i,j}$ は, このような確率的戦略を用いた場合の “ C ” (協力) を出す確率 (その収束値) である。無限繰り返しゲームを行う場合に “協力する確率” は一定に収束するのでその場合の期待利得を計算した, ということである。Nowak and Sigmund (1990) 参照。

5) polite, tolerant という態度は Axelrod (1997) による命名である。

6) このフレームワークに対しては, 学会発表時に次のような指摘があった。遺伝的アルゴリズムでは次世代の遺伝子型を作り出す際に “親” となった 2 個体の遺伝子型が “交叉” を行うが, このフレームワークでは一つの遺伝子型が 2 つの表現型 (実数) を生み出しているため, 集団の中での性質の分布が世代間で極端に変化してしまうのではないかというものである。補論で述べるように, 指摘されたような事態がおきてもこの分析では大きな問題を引き起こさないが, この指摘は遺伝子型のコーディングそのものに対して注意を払う機会となった。学会報告時に問題提起をして頂いた参加者の方々に感謝する。

- p, q に対してそれぞれ長さ 10 の 2 進数文字列を割り当て、それをつないだ長さ 20 の 2 進数文字列を遺伝子型とする。
- 遺伝子型の前半と後半の 2 進数文字列をそれぞれ $[0, 1)$ の範囲の実数に変換して表現型 (p, q) とする。
- 遺伝子型の文字列 1 つを 1 個体とみなし、100 個体からなる集団を作る。1 集団、または 2 集団でのランダム対戦で無限繰り返しゲームをおこない、(1)式から得られる適合度にしながら、遺伝的アルゴリズムを実行する⁷⁾。

本稿においても、これと同じフレームワークを使用する。

3 1 集団での戦略推移

3.1 集団の状態

今回のシミュレーションでは、100 個体からなる集団を初期生成 (世代 = 0) し、

- ゲームの実行による各個体の適合度決定
- 適合度に基づく“親”の選択
- 交叉と突然変異

からなる遺伝的アルゴリズムをへて次世代集団 (世代 = 1) を決定している。

この過程で適合度が低い個体は淘汰され、より高い適合度を得た個体の遺伝子型が集団内に広まっていくのであるが、遺伝子型の変化に伴い表現型である (p, q) の値も変わっていくため、各世代でそのときの集団の個体分布を記録

7) 交叉確率 0.25, 突然変異率 0.001。一点交叉を用いて、エリート戦略 (その世代で最大適合度をあげた個体の遺伝子型はそのまま次世代へ受け継がれる) を使う。また適合度については線形スケーリングを行っている。

することで集団の状況 (p, q 平面上での個体分布) を観察することができる。以下では、100 個体からなる 1 集団が集団内で相互にランダム対戦を行い、その結果によって集団が進化していく過程を個体分布から観察する。なお、遺伝的アルゴリズムが通常使われる関数の最適解を探す場合と異なり、このシミュレーションではアルゴリズムが続く限り遺伝子型は変化していく。そこで今回のシミュレーションでは、世代数の上限を 1,000 世代に統一した。

集団が全体としてどのような状態にあるのか、いわばマクロ的な状況は、各個体が獲得した適合度の集団平均値によって観察できる。ここで注目する状態は、各世代での集団内のゲームにおいて多くの個体が (C, C) を実現している状態である。ここでは集団内の 8 割の個体が各ゲームで (C, C) を実現した状態を“集団が協力へ収束した状態”、逆に 8 割が (D, D) となっている状態を“集団が裏切りへ収束した状態”と定義する。

そうした状態の集団の平均適合度は、“協力状態”では 2.76 以上、“裏切り状態”では 1.56 以下になっている⁸⁾。世代の進行中にたまたま平均値がこうした基準を超えることはあっても、それが持続しない限り「集団が協力へ（裏切りへ）“収束した”」とはいえない。ある程度そのような状態が持続していることをもって、“収束”を定義しよう。ここでは「連続した 10 世代中の 6 世代以上で集団平均適合度が 2.76 以上 (1.56 以下)」となっている状態を「集団が協力へ（裏切りへ）収束した」と定義する。また、 (p, q) 正方形を安田俊一 (2005) と同様に以下に区分し、全体としての集団の性質を考える。

あまのじゃく領域 ($0 \leq p < 0.5, 0.5 \leq q \leq 1$) にある個体は、報恩も報復

8) 協力状態においては 8 割の個体が C 戦略、2 割の個体が D 戦略をとっている。 C 戦略をとる個体の期待利得は $0.8 \times 3 + 0.2 \times 0 = 2.4$ 、 D 戦略をとる個体の期待利得は $0.8 \times 5 + 0.2 \times 1 = 4.2$ 。したがって、集団の平均利得は $0.8 \times 2.4 + 0.2 \times 4.2 = 2.76$ 。同様に、裏切り状態においては 2 割の C 戦略者と 8 割の D 戦略者がおり、 C 戦略をとる個体の期待利得は $0.2 \times 3 + 0.8 \times 0 = 0.6$ 、 D 戦略をとる個体の期待利得は $0.8 \times 1 + 0.2 \times 5 = 1.8$ 。したがって、集団の平均利得は $0.8 \times 1.8 + 0.2 \times 0.6 = 1.56$ 。

の程度も弱い。相手の協力に対して応えず、裏切りに対しても報復しない傾向がある。

裏切り領域 ($0 \leq p < 0.5$, $0 \leq q < 0.5$) にある個体は報恩の程度が弱く、報復の程度が強い。相手の手に関わらず裏切る傾向にある。

お人好し領域 ($0.5 \leq p \leq 1$, $0.5 \leq q \leq 1$) にある個体は報恩の程度が強く、報復の程度は低い。相手の手に関わらず協力する傾向にある。

互恵主義者領域 ($0.5 \leq p \leq 1$, $0 \leq q < 0.5$) にある個体は、報恩の程度も報復の程度も強い。前回の相手の協力に対しては協力で、裏切りに対しては裏切りで報いる傾向にある。TFT 個体 ($p=1$, $q=0$) はこの領域に含まれる。

3.2 戦略の推移例

図1は、あるシミュレーションにおける1集団の平均適合度の推移をみたものである。矢印の各数字は世代数、図中の上方のラインが集団の平均適合度 = 2.76 のラインである。集団の平均適合度がこのラインを超えていればこの集団が協力状態を達成していることがわかる。

図2は図1中に示した各世代での個体分布を示している。それぞれの分布図の右上にはそのときの集団の平均 p , q および平均適合度の大きさが示されている。

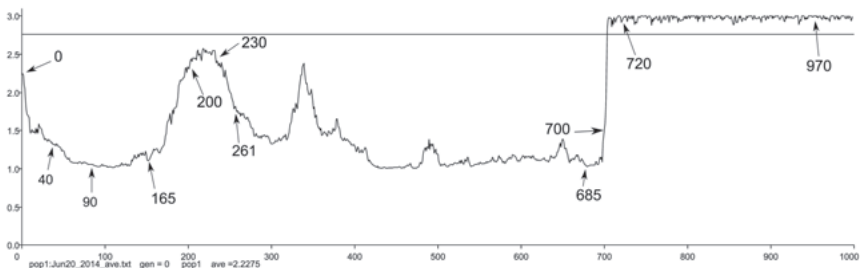


図1 1 集団ゲームでの典型例

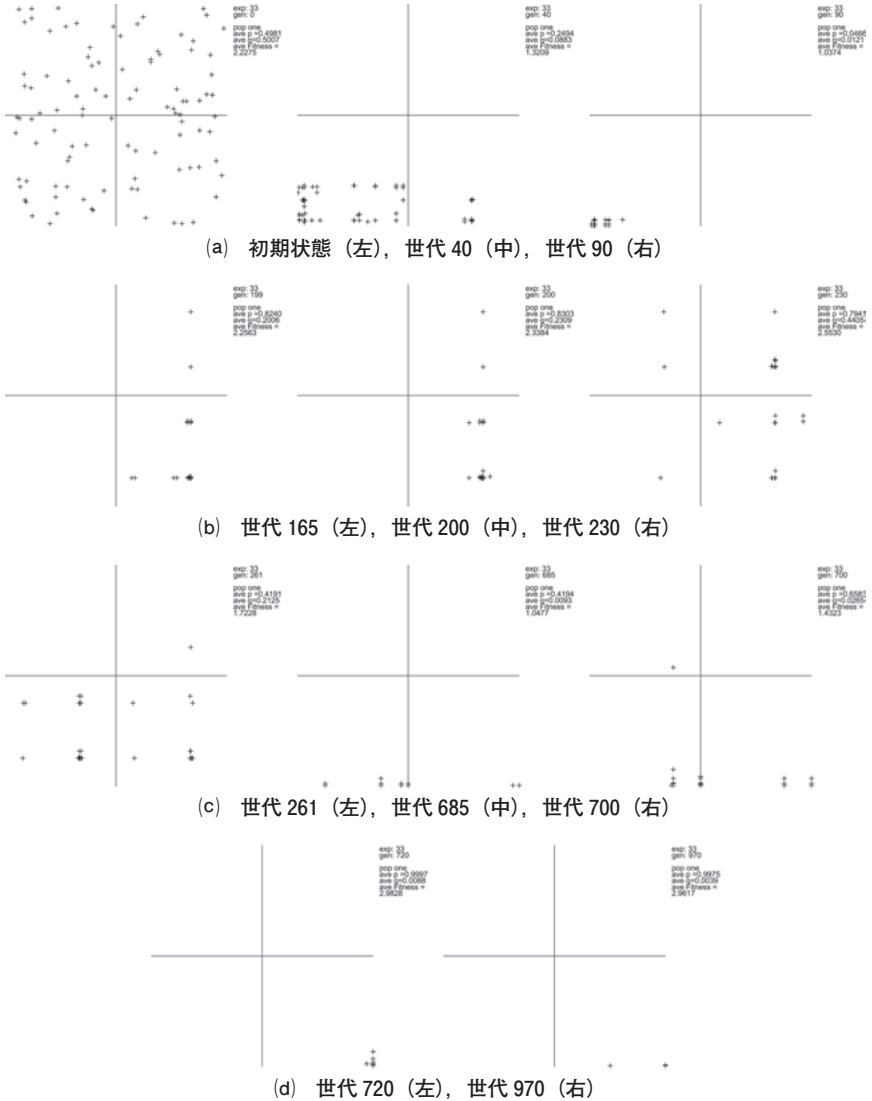


図 2 各世代での分布状態

図で示したシミュレーションでは、初期に発生した集団はランダムに個体が分布している初期状態（図2(a)左）から以下のような変遷をたどっている。

1. ランダムに分布している集団は、はじめに報復が強まる傾向へ進化し（40世代）、90世代にはほぼすべての個体が裏切り（ $\bar{p}=0.05$, $\bar{q}=0.01$ ）（ \bar{p} , \bar{q} は表現型の集団平均値）へ進化する。集団の平均利得は1.04であり、ほぼすべての個体が裏切り戦略者となる（図2(a)）。
2. こうした状態がしばらく続いた後、165世代あたりでは互惠主義者領域に個体がある程度発生している。集団内で互惠主義者と対戦した個体は、“協力”を得た場合は高い利得を得ることができるため、 p が高い個体ほど適合度を上げ、集団内に広がっていく。それに応じて集団適合度の平均値が上がり始める。200世代には報恩の程度が強まり、平均値がかなり高くなっている。
3. しかし、このときには完全に協力へ収束することはなかった。230世代では“報復”が弱いお人好し領域、あまのじゃく領域に個体が発生している。また、裏切り領域にも、わずかに個体が発生している。こうなると、わずかに発生した裏切り戦略者はお人好し戦略者やあまのじゃく戦略者との対戦において高い利得（ $T=5$ ）をえることができる（図2(b)）。
4. 261世代では、報復の程度が高い領域に多くの個体が発生しており（ $\bar{q}=0.2$ ）、裏切り領域にも多くの個体が発生してきている。その結果集団の平均適合度は低くなっている（図2(c)左）。
5. 変化はあるものの、この状態が持続し、しばらくは集団の平均適合度は低いまま推移する。
6. こうした中で685世代にごくわずかの TFT 個体が発生した。後に述べるように TFT 個体との対戦は集団を協力へ向かわせる傾向がある。ここでも急速に協力への傾向が強まっていき、700世代には集団全体の

報恩が強まる ($\bar{p}=0.658$) (図 2(c)中, 右)。

7. この過程がつづき, 720 世代には集団がほぼ TFT に固まり, 集団は協力へ収束した。この場合は, 上記 3 の状態と異なり, 報恩も報復も共に強い個体がほとんどである ($\bar{p} \simeq 1.0$, $\bar{q} = 0.01$)。この状態は 970 世代でも同じでほとんどが TFT であるため, 協力状態がそのままシミュレーションの終わりまで持続した (図 2(d))。

このように個体分布の状態と集団の平均適合度の高さはある程度関係している。安田俊一 (2005) で分析したように, 協力を達成できる集団においては報復の程度が強い傾向を持つ個体の比率が集団の平均適合度に強い影響を持っており, 協力を達成できない集団においては報復の程度が弱い傾向を持つ個体の比率, 特に「お人好し」領域の個体比率が集団の平均適合度に影響を持つ。図 2 での個体分布は, 一回のシミュレーションにおける時系列変化の中でも同様の傾向があることをしめしている。

このことは「協力が崩壊したケース」をみればよりはっきりする。

3.3 協力の生成と崩壊

1 集団ゲームにおける協力の生成と崩壊の例を図 3 に示す。これは 1 回のシミュレーションの中で, 協力への未達 (275 世代), 協力状態の持続 (450, 545

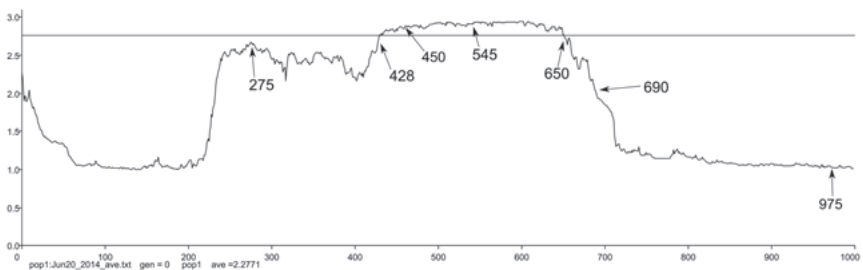


図 3 1 集団ゲームでの協力の生成と崩壊

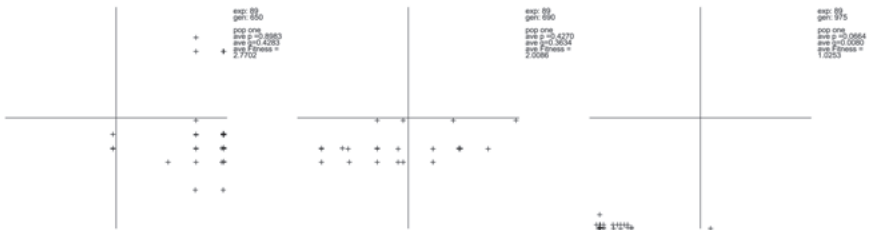
世代), 協力の崩壊 (650 世代), 裏切りへの収束と持続 (690, 975 世代) が見られた例である。それぞれの世代での個体分布を図 4 に示す。

275 世代 集団の平均適合度は 2.67 であり, 協力へ収束しそうな傾向であるものの, 収束はしなかった。この時点ではお人好し領域に個体があり, また, $\bar{q} \approx 0.4$ で報復の程度がそれほど強くない (図 4(a)左)。

428 世代 協力へ収束する直前の状態。平均利得 2.77 でほぼ 275 世代と同じであるが, いくつかお人好し領域に個体が残っているものの, ほとんどの個体が互惠主義者領域に集中している。また, 275 世代より報復の程度が強く, 報恩の程度も強い ($\bar{q} = 0.275$, $\bar{p} = 0.9$)。特に右下にはほぼ TFT 個体が現れている。この状態からは協力への収束が達成される (図 4(a)中)。



(a) 協力の未達 (左), 協力の生成 (中), 協力の持続 (右)



(b) 協力の崩壊 (左), 裏切りへの遷移 (中), 裏切りの持続 (右)

図 4 協力の生成・崩壊における分布状態

450 世代, 545 世代 協力へ収束し, その状態が持続している。この状態を, おなじく協力への収束が持続している図 2(d)と比較してみると, あきらかに報復の程度がことなっている。図 2(d)の 720, 970 世代ではそれぞれ $\bar{q}=0.008, 0.004$ であるのに対して, 450 世代, 545 世代では $\bar{q}=0.275, 0.294$ と, 報復の程度が比較的低い。

650 世代 そうした低い報復の程度はお人好し領域に個体がある程度発生させる。すると裏切り戦略者が高い利得をえる機会が増える。その結果集団の報恩の程度も弱くなる方向へ集団が拡散しつつある状態が観察できる (図 4(b)左)。

690 世代, 975 世代 結局, この協力への収束状態は崩壊し, 集団の個体分布は裏切り領域へ向かっている。そのまま, 裏切りへの収束状態となり, 集団は報復が非常に強く, 報恩の程度が弱い状態を続けた(975 世代: $\bar{p}=0.066, \bar{q}=0.008$) (図 4(b)中, 右)。

以上の観察を元に 1 集団におけるゲームでの結論をまとめると以下のようになる。

1. 集団は報復軸に関しては, それが強まる傾向へまず進化する。分布で言えば, 初期状態からまずは下向き (q が低くなる方向) へ偏る。これはほぼすべてのシミュレーションにおいて, 初期には集団の平均適合度が低下することから読み取れる。囚人のジレンマでは (D, D) が Nash 解であることから, ランダムな初期状態からの対戦ではまずはじめにそれが実現することを意味している。
2. 協力への進化は集団中に報恩の程度が高い個体が発生することにより引き起こされる。報恩の程度が高い個体は, 同じように報恩の程度が高い個体との対戦で高い適合度を得るため, 集団全体で報恩の程度が強くなる。そうした個体が集団内で増えていけば集団の平均適合度が高くなる。

なっていく。

3. 協力が持続するかどうかは、集団の報復の程度に関係がある。全体として報復の程度が強い場合は協力が持続し、そうでない場合は協力は長続きしない。集団が互惠主義者領域に固まったとしても、個体分布が比較的“縦方向”に伸びている場合には、 D に対して C で応答する確率が残っていて、そうした個体に低い適合度を与えられ、 D 戦略者が拡大するきっかけを与えてしまう。
4. おなじように、協力状態への未達成にも報復の程度の低さが関係している。協力状態へ達成しない場合は、お人好し領域に個体が発生していることが観察されることから、同じ理由で協力状態へ達成する前に D 戦略者が拡大する可能性が高い。

つまり、集団が協力状態へ至り、それを持続するためには、“報恩と報復”の両方のシステムが備わっている必要があることがわかる。この点で Axelrod (1984, 1997) が強調した“しっぺ返し：Tit For Tat, TFT”戦略はそれが進化的に安定かどうかというよりは集団を協力状態へ向かわせるトリガーとなっていることが、これらのシミュレーションからわかる。

4 2 集団におけるゲーム

異なる集団の間でゲームを行うと、1 集団でのゲームではみられなかった状態が現れる。

先に見たように、報恩の程度が強いけれども、報復の程度は弱いお人好し領域の個体は、裏切り戦略者との対戦では (T, S) もしくは (S, T) が実現してしまい、利得が 0 となる。

1 集団内でのゲームの場合には、そうした個体は淘汰されて短時間で消滅するし、同じ集団内に $T=5$ の適合度を得る個体が存在するため、集団の平均適合度としてはそれほど低くならない (図 1, 230-261 世代)。

しかし、2 集団で行われるゲームの場合には片方の集団に“お人好し”が集まり、他方の集団に裏切り戦略者が集まる場合がある。その場合には裏切り戦略者からなる集団の平均適合度は高くなり、お人好しからなる集団の平均適合度は低くなる。裏切り戦略者からなる集団はお人好しからなる集団を、いわば“えさ”にしている状況が発生する。このような状態をここでは“搾取”と呼ぼう。一方の集団（これを集団 1 とする）が他方の集団（集団 2）を「搾取する」とは、囚人のジレンマゲームにおいて、集団 1 の多くのメンバーが集団 2 の多くのメンバーとの対戦で、(T, S) の状態にある、と定義する。搾取が起きている場合のメルクマールは、集団 1 の平均適合度が 3.84 を超え、かつそのときに集団 2 の平均適合度が 0.84 を下回ったときとする⁹⁾。また、1 集団の場合と同じように 2 集団における協力状態の定義は両集団共に集団平均適合度が 2.76 を超えることである。同様に裏切りへの収束は両方の集団が 1.56 を下回ることである。

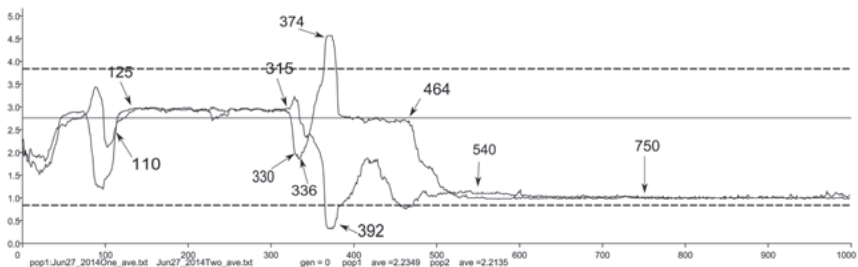


図5 集団 1 による“搾取”

9) 集団 1 の 8 割のメンバーが「D」を出し、集団 2 の 8 割のメンバーが「C」を出しているとしよう。そのとき、集団 1 のメンバーの期待利得は、D を出す個体が第 2 集団にいる 8 割の C 戦略者と 2 割の D 戦略者と対戦するため、 $0.8 \times 5 + 0.2 \times 1 = 4.2$ 、C を出す個体が $0.8 \times 3 + 0.2 \times 0 = 2.4$ 。集団 1 には D が 8 割、C が 2 割だから、集団 1 の平均期待利得は $0.8 \times 4.2 + 0.2 \times 2.4 = 3.84$ 。

一方、集団 2 のメンバーの期待利得は、C を出す個体が第 1 集団の 2 割の C 戦略者と 8 割の D 戦略者と対戦するため、 $0.2 \times 3 + 0.8 \times 0 = 0.6$ 、D を出す個体が、 $0.2 \times 5 + 0.8 \times 1 = 1.8$ 。集団 2 には C が 8 割、D が 2 割だから、集団 2 の平均利得は $0.8 \times 0.6 + 0.2 \times 1.8 = 0.84$ となる。図 5 では、そのラインを点線で示している。

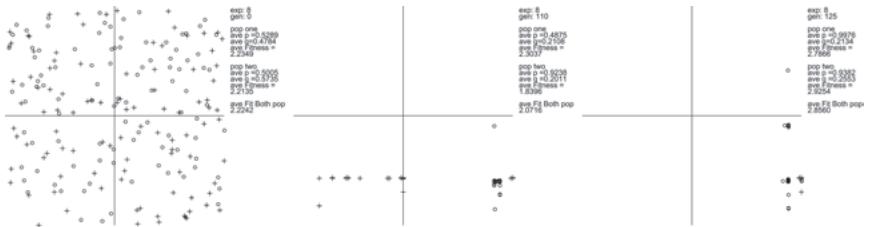
図5は、2集団におけるゲームでの両集団の平均適合度の推移を描いたものである。図中に示した特徴的な世代における個体分布を図6に示す。分布図中、第1集団は「+」印、第2集団は「○」印で個体を表す。このシミュレーションでの推移の特徴は以下である。

初期状態からの推移 上述のように、初期状態でランダムに生成された集団は、最初は報復の程度が強まる方向へ進化する。2集団でのゲームでも、しばらくは両集団の平均適合度が低下する方向へ向かう。このシミュレーションの場合には、その後、いったん第1集団が第2集団を“えさ”にすることで第1集団の平均適合度があがるが、“搾取”にまではいたっていない。

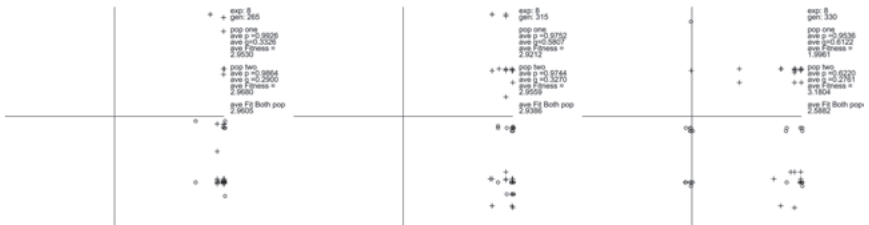
協力状態への移行 第110世代では第2集団が互惠主義者の領域に“ある程度の集団”（以下、「コロニー」）を作っており、それらとの対戦を通じて第1集団も互惠主義者領域にむかって、報恩を強める方向へ進化した。そして第125世代では両方の集団が平均適合度2.76を超えて協力状態に達した。しかし、このときの個体分布をみると、集団2は報復の程度が弱い個体が残っており（図6(a)右。集団2の $\bar{q}=0.266$ ）、第1集団の時と同じく、協力を崩壊させる原因を内包している。

協力状態の崩壊 協力が持続している間に、こんどは第1集団がより報復の程度が弱い方向に進化した（図6(b)左。集団1の $\bar{q}=0.333$ ）。その結果、集団2がより報恩が弱い状態へ移行し（第330世代、集団2の $\bar{p}=0.622$ ）、協力は崩壊した（両集団の平均適合度2.58）。

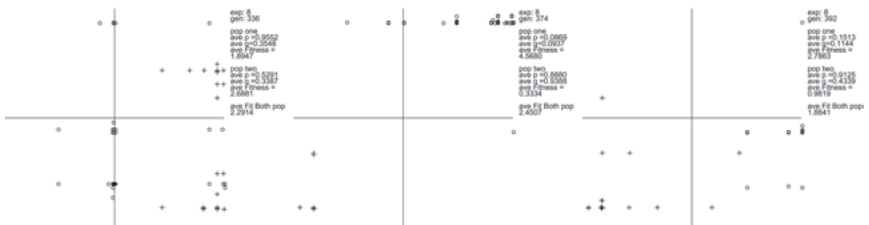
集団2の悪化と集団1の反転 第336世代では集団2は引き続き、報恩が弱い状態へ移行し（ $\bar{p}=0.53$ ）、集団1は報恩が高い状態に移行している（ $\bar{p}=0.955$ 、図6(c)左）。このとき、集団1は互惠主義者領域とお人好し領域にコロニーがあり、どちらかと言えばランダムプレイヤ



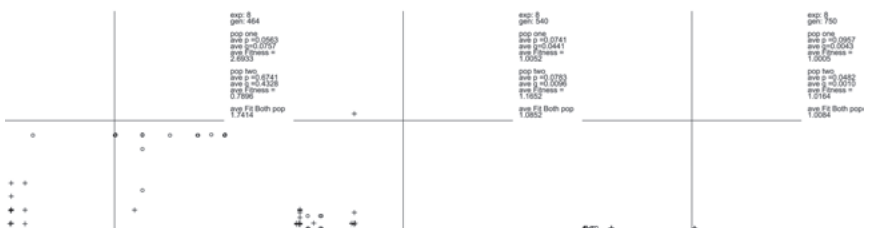
(a) 初期状態 (左), 協力の生成 (中), 協力の達成 (右)



(b) 協力状態 (左), 協力の崩壊 (中), 集団 1 の悪化 (右)



(c) 集団1の反転 (左), 集団1による集団2の搾取 (中), 集団2の反転 (右)



(d) 裏切り状態への集中 (左), 裏切り状態 (中), 裏切り状態への収束とその継続 (右)

図6 2 集団における協力の生成・崩壊における分布状態

一に近い状態となった集団2との対戦で高い利得を得始めている。集団の平均適合度は第2集団で下がり始め、第1集団は上昇を始めた。

集団1による集団2の搾取 この傾向がつづいて、第374世代には第1集団が第2集団を搾取する状態に達している。図6(c)中図にこのときの個体分布が描いてある。これをみると、第2集団がお人好し領域にコロニーを作り、裏切り領域にコロニーを作っている第1集団の“えさ”になっている状況が明確である。第1集団は4.57という高い平均適合度をあげ、第2集団の平均適合度は0.333と低い。

集団2の反転 この状態から、第2集団はより報復の程度が高い方向へ進化を始め、互惠主義者領域に個体が発生してくる（第392世代、6(c)右）。それにより、集団2に平均適合度は増加しはじめた。

裏切りへの集中開始 集団1は集団2がより報復の程度が高い方向へ進化していく間にいったん報恩が高い領域にも個体を発生させた。第392世代からおおよそ第450世代では第2集団の平均適合度がいちど高くなり再びさがって“山”を作っている世代がそれにあたる。その後、第1集団はほぼ裏切り領域に集中した。それに対応して第2集団も裏切り領域へ進化している（第464世代。図6(d)左）。

裏切りへの収束とその持続 最終的に、このシミュレーションは両集団とも裏切り領域に個体が集中し、ほぼ完全に裏切り状態への収束状況が続いた（第540世代、第750世代。図6(d)中、右）。

以上が典型的な2集団によるゲームでのシミュレーション概要である。

1集団によるゲームと2集団によるゲームを観察すると、集団が協力状態を達成するきっかけとなるのは、互惠主義者領域への個体の発生であることがよくわかる。

1集団のゲームでは、互惠主義者領域に発生した個体が高い利得をあげると、その遺伝子型は交配を通じて集団内で広まっていく。しかし2集団の場合

には相手集団でそうした個体が発生した場合には交配による遺伝子型の拡散はない。したがって同一パラメータで行ったシミュレーションにおいても2集団でゲームを行う方が、協力へ達成する頻度は低い。

協力を達成したシミュレーションの数を比較してみると、その違いは明白である。今回のシミュレーションでは1,000世代中に一度でも協力状態を達成したシミュレーションが100回になるのに、全体で何回のシミュレーションが必要であったかをカウントした。その結果、1集団ゲームでは協力へ収束するシミュレーションを100回生成するのに376回のシミュレーションが必要であったのに対して、2集団ゲームでは3,159回必要であった。2集団のゲームでは協力を達成するのは1集団のおおよそ10倍難くなる(表2)。2集団ゲームにおいても互惠主義者領域に発生した個体が高い利得を得た場合にはその遺伝子型が集団内に拡散していくが、そうなるためには他方の集団にお人好し領域か互惠主義者領域にある程度の個体が必要なのではない。このことが2集団ゲームにおける協力達成の困難をもたらしている。

上述のようにTFTは協力達成へのトリガーとして機能する。安田俊一(2001)では、TFT個体との対戦確率と各領域で発生した個体数の相関を分析したが、ここでは集団の協力状態への達成そのものについて見てみよう。

5 TFT 効 果

本稿での設定において、TFT個体は「完全に報恩的で完全に報復的な個体の表現型」 $(p, q) = (1, 0)$ で表される。(1)式から、TFT戦略者との対戦では、任意の表現型 (p_i, q_i) を持つ個体の期待利得は

$$A^i(E^i, \text{TFT}) = 1 + 3c - c^2, \quad c = \frac{q_i}{1 - (p_i - q_i)} \quad (2)$$

となる。

各世代である個体がTFTと対戦する確率を“TFT濃度”と呼び、TFTと対

TFT 濃度	1 集団ゲーム	2 集団ゲーム
	必要シミュレーション回数	必要シミュレーション回数
0 %	376	3,159
1 %	367	2,085
2 %	264	1,781
3 %	267	1,073
4 %	244	946
5 %	228	643
10%	170	259
15%	127	162

表1 100回の協力達成を得るために必要だったシミュレーション回数

戦することになった個体は(2)式に従った利得を得るものとする。

この条件で1,000世代中に協力を達成するシミュレーションを100回得るのに必要な回数をカウントした結果が以下の表である。表から明らかなとおり、TFTが協力の達成に与える影響は大きい。特に2集団ゲームにおいては顕著である。ここでのTFT戦略者は遺伝子型で定義されたものではなく、単に(2)式に従って対戦相手の適合度を決定しているだけなので、なんらかのTFT戦略者としての行動を担う遺伝子型を集団内に拡散しているわけではない。したがって、集団はTFT戦略者から与えられる適合度に対応する形で協力を達成するように進化の方向を決めている。

安田俊一(2001)で示したように、TFT戦略者との対戦では報恩の程度が高い個体(互惠主義者領域とお人好し領域)に対して高い適合度を与える。その結果、集団の中にそうした性質を持つ個体を発生させていく。すでに述べたように集団の中にそうした個体が発生した場合には、集団全体が報恩の程度を高めていく。

この結果は現実社会を解釈する際の、また、集団を協力へ向かわせる社会システムを考察する際のポイントの一つとなる。

6 結語：現実社会の解釈

囚人のジレンマを表現する利得表では、 (D, D) が Nash 解であり、その結果 (P, P) の利得が実現する。これは合理的な 2 人のプレイヤーでこのゲームを行う限り必然的な帰結である。しかしながら、現実社会を考えると社会全体が常にそのような状態になっているとは必ずしもいえない。社会全体としては時として、部分的にはより頻繁に (C, C) 解が成立している。そうでなければ長期間社会が持続することはできないはずである。

個人合理性から考える限り成立しそうにない状態が、社会全体になると完全ではなくとも現実存在する。この理由のひとつが、社会の状況は集団を構成する個体の分布により決定されるという事実である。今回のシミュレーションで焦点を当てたのは、そうした個体分布と集団の状況とのダイナミクスの観察であった。上述のようにシミュレーションでは“協力状態”や“協力状態への収束”という言葉を使っているが、これを「プレイヤーの戦略として C 戦略が進化した」ととらえるのは、その意味で誤りである。あくまで個体の分布として全体の特徴が生まれ、そのことが“集団の平均適合度”という尺度で表現された集団（社会）の状態を決定していることを示しているのである。

“報恩”と“報復”は人間社会の中で古くから“教訓”や“処世術”として引き継がれてきている考え方の軸である。Nowak たちが示した分析のフレームワークはこの点で人間の社会行動を考察する上では非常にシンプルで本質的な部分をとらえているといえよう。

一連のシミュレーション結果から考えると、囚人ジレンマゲームをベースとするような社会においては、集団は基本的に“より報復的な”個体分布を持つ。“より寛容”な領域である“あまのじゃく”、“お人好し”領域に個体がある程度発生しても、そのことは「より報復的でより報恩的でない」個体の発生を促し、“より寛容”な領域の個体は淘汰される。むしろ、そうした領域にある程度の個体が発生することが“裏切り領域”の個体を増加させているともいえる。

この結果は、社会が寛容でありすぎればそれを利用して自己の利益をあげる個体が増加し、不寛容な社会を生み出すという逆説的な結果をもたらすことを示唆している。こうしたことは現実の社会や組織でいかにもありそうなことであり、実際に体験上からも多くの人が首肯できよう。また2集団ゲームのように、互いに交流がない集団間でゲームが行われる場合には、片方の集団がより寛容な集団を一方的に犠牲にする“搾取”の状態が起きることも現実味のあることである。

また、シミュレーションの結果から観察できるのは、以上のように報復的な傾向が強い個体分布が基本である集団が協力状態を達成するには報恩の程度が強い個体の存在が重要な役割を果たしていることだ。報恩の程度が高い個体がある程度集団の中に発生すると、そうした個体との対戦で高い利得を獲得する、やはり報恩の程度が高い個体により多く発生し、集団は協力状態へ引き寄せられる。

遺伝的アルゴリズムでは遺伝子型が次世代に遺伝することで集団の状況が変化していく。人間社会では、個人の体験という“遺伝子型”は生物学上の遺伝子のように物理的な交配によって情報が伝えられるのではなく、“学習・模倣”といった手段で伝えられていくことを考えると、報恩的な個体の行動は、“模範”として、あるいは“ノウハウ”として社会の中に伝播していくと考えることができる。それらが歴史的には“物語”として社会に蓄積されていき、また、“モラル”として定着していく。

そう考えると、逆に、社会が (D, D) 状態に陥ってしまわないような装置として、“恩に報いる物語”や“報恩を美德とするモラル”が発生してきたとも解釈できる。

この点、ここで示したように、TFT戦略がそうした戦略をもつ個体としてではなく、いわば集団の外部から利得を与える形で投入されたシミュレーションが協力状態を発生しやすくしていることは大変興味深い。

現実社会においては、例えば法制度のように、人間関係の外にあって、それ

から生ずる結果に対して規範的に判断をくだし、ときとして罰を与える存在は、このシミュレーションにおける TFT “個体” と同じような役割を果たしているとも解釈できる。ただし、法制度は基本的に報復の為の装置であるから、社会全体が報復的な傾向をもつように誘導するが、報恩の程度には影響をあたえていないかもしれない。このことは、あまりに厳罰的な法制度でがんじがらめにされた社会は、むしろ“裏切り状態”的な社会をもたらすのではないかという、おそらくは多くの人々が持っているであろうイメージと合致する。

そのように考えると、社会や組織が協調的な性質を獲得・維持して、それらを構成する主体が (C, C) というパレート最適な状況を達成するために必要なのは、“報恩”を何らかの形で制度化することではないか。

社会や組織のルールから逸脱した個体に報復するしくみは様々な形で制度化され、ひろく運用されているが、ルールを遵守した者に対する積極的な報恩の仕組みは制度化も運用もそれほどなされていないように思われる。具体的な制度の検討は本稿の考察範囲を超えるが、コンピュータシミュレーションの結果がこうした社会的な含意を持つ形で解釈できることを示し得たことは一連の研究の成果であろう。

A 遺伝子型のコーディングについて

本稿のシミュレーションのように遺伝子型を2進数としてコーディングした場合、交叉によって表現型が大きく変化してしまうのではないかと、との指摘を学会発表時に受けた。後述のように、本稿の分析方法においては交叉による表現型の変化がシミュレーション結果に大きな影響を与えることはないが、この指摘によって遺伝子型のコーディングについて検討する機会を得ることができた。

以下では遺伝子型のコーディングに関わる問題について簡単に考察する。

A.1 交叉の問題

学会発表時に指摘された問題は以下のようなことである。

いま、 A 、 B 2つの個体の染色体が、それぞれ 20 ビットの文字列としてコーディングされ、前半 10 ビットを p 、後半 10 ビットを q に対応させるとする。

$$\begin{array}{l}
 A : \overbrace{a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10}}^p \underbrace{a_{11} a_{12} a_{13} a_{14} a_{15} a_{16} a_{17} a_{18} a_{19}}_q \\
 B : b_0 b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10} b_{11} b_{12} b_{13} b_{14} b_{15} b_{16} b_{17} b_{18} b_{19}
 \end{array}$$

ここで、 $a_i, b_i \in \{0, 1\}$

たとえば 13 ビット目と 14 ビット目の間で B との交叉がおきると、交叉後の染色体のひとつは

$$a_0 a_1 a_2 a_3 a_4 a_5 a_6 a_7 a_8 a_9 a_{10} a_{11} a_{12} b_{13} b_{14} b_{15} b_{16} b_{17} b_{18} b_{19}$$

となるため、遺伝子型を表現型に変換したときに大きく値が変化してしまう。それによって、個体分布が交叉のたびに大きく変化してしまうのではないかというのが指摘された問題である。

一般に 2 進数として遺伝子型を表現し、それを整数変換することによって表現型とする場合、交叉後は値が大きく変化するように思える。例えば 10 ビット表現での (0000001010) (整数の 10) と (0000111011) (整数の 59) が 7 ビットと 8 ビットの間で交叉した場合、前者は (0000111010) となり、整数の 58 を表すことになる。

一つの遺伝子型 (今の例では最初の (0000001010)) だけに着目している限りではその通りであり、交叉により表現型が大きく変化することになる。この例では元の遺伝子型が示す整数値は 10 で変換後は 58 へと大きく変化した。

しかしこの例でも、もう一方の遺伝子型 (0000111011) は、このときの交叉

後に (0000001011) となっていて、これは整数 11 である。つまり、このケースでは 2 つの遺伝子型は“立場を入れ替え”ていて、表現型でいえば (10, 59) であった組が交叉後には (58, 11) となっている。交叉した 2 つの遺伝子型を“一組”と考えれば、前後の値が入れ替わったことになる。

これは 2 進数で表された遺伝子型を、整数変換して表現型とする方法をとる限り当然のことである。つまり 2 進数から 10 進数へ変換する際の通常のルールでは上位ビットになるほど整数の中にしめる影響が大きいので、片方の染色体から上位ビットが他方へうつと、移った先でのそのビットの影響力は大きい。そのため、多少の違いはあっても、表現型としては 2 つの 2 進数が表す 10 進数が入れ替わったにすぎないことになる。

ここでのコーディングでは上記のように 20 ビットのバイナリ文字列を前半と後半に分け、長さ 1 の正方形における座標 (p, q) を表現型としている。個体 i の表現型を (p_i, q_i) 、個体 j の表現型を (p_j, q_j) とすると、前半 10 ビットの中に交叉ポイントがある場合には、後半 10 ビットが表す q_i, q_j は変化しないので、 q 軸座標は交叉後も変化しない。

変化は前半 10 ビットが表す p_i, p_j で起きるわけだが、上述のように多少のずれはあるにしても p_i, p_j は立場を入れ替えるだけである。交叉前が (p_i, q_i) 、 (p_j, q_j) の 2 点であったのが、交叉後には (p'_j, q_i) 、 (p'_i, q_j) になり、 $p'_i \simeq p_i$ 、 $p'_j \simeq p_j$ なのだから、平面上の位置で言えば、2 点は縦軸座標 (q 座標) は変わらず、横軸座標 (p 座標) の値が少々ずれるだけになる。交叉ポイントが下位 10 ビットの中に入ったとしても、今度は p_i, p_j が変化せず、 q_i と q_j がほぼ入れ替わっただけになる。

ここでは一つの遺伝子型が平面上の点を表現している。2 つの個体が交叉を行えば、上の理由でその 2 つの個体は平面上の位置を入れ替えることになる。交叉後には中の点がそれぞれ交叉前の位置と入れ替わっているだけなので、集団としてみれば、分布がそれほど大きく変わっているわけではない。特に、この分析のフレームワークでは 10 ビット 2 進数で表される 10 進数を区間 (0, 1)

にマッピングして表現型としているため、交叉による表現型の変化は集団全体としてみればとても小さくなる。

したがって、個体を平面上の点としてとらえているかぎり、交叉は個体分布にあまり大きな影響を与えない。

今回のシミュレーションにおいては遺伝子型の設計を変更し、個体の表現型(p, q)を担う遺伝子型をそれぞれ12ビットの2進数でコーディングしてみた。このコーディングでは平面上の点で示される個体の p, q 座標のそれぞれが交叉するため2点がほぼそのまま場所を入れ替えることになる。この変更が集団の進化に与える影響はほとんどなかった。表2に「1染色体」とあるのが、20ビットの2進数を前半と後半に分けてそれぞれ p, q としたコーディングによる結果であり、「2染色体」とあるのが、上記のように p, q をそれぞれ12ビットの2進数でコーディングしたものである。

2集団ゲームにおいてやや協力を進化させやすいような結果も出ているが、1集団ゲームでは逆に協力が進化しにくくなっていて、明確な違いは検出できなかった¹⁰⁾。

1 染色体	1 集団ゲーム	2 集団ゲーム
通常 2 進数	376	3, 159
Gray code	134	231
2 染色体	1 集団ゲーム	2 集団ゲーム
通常 2 進数	400	2, 467
Gray code	129	245

表2 協力へ収束する100回シミュレーションに要した回数

10) もちろん、厳密に違いを検定しようとすれば、こうした実験を大量に行い、統計的な検定を行うべきであるが、コーディングを変えた実験の趣旨は、本文に記した結論に影響を与えるほどの顕著な違いがないことのみを確かめるためだけなので、これで十分である。さらに本文で述べたとおり、「点の入れ替わり」ということではそうたいした違いはないため、そのような検定の必要はないものと判断した。

集団の協力への進化に対して遺伝子型の設計という視点から大きな影響をもつのは、遺伝子型から表現型への変換ルールである。

以下ではその点を検討する。

A. 2 Gray coding

本稿では遺伝子型を 0, 1 を要素とする文字列でコーディングしているが、遺伝子型から表現型への変換ルールについては、「文字列を通常の 2 進数とみなす方法（通常 2 進数）」と「文字列を Gray code による 2 進数とみなす方法（Gray code）」の 2 つの方法でシミュレーションを行った。いずれの方法も、いったん遺伝子型を整数変換し、区間 $[0, 1)$ にマッピングするが、通常 2 進数と Gray code では 10 進数への変換方法が異なる。

10 進数「10」を通常の 2 進数で表すと (1010)、10 進数「9」は 2 進数表現では (1001) となり、10 進数で 1 だけ異なる数値が、2 進数表現では 2 カ所異なっていることがわかる（「ハミング距離」が 2）。

一方、Gray code を用いると、「10」が (1001)、「9」が (1011) であり、違いは 1 カ所しかない（「ハミング距離」が 1）。

このように、Gray code は隣接する数値のハミング距離が常に 1 となるような変換ルールを持つ。

なんらかの最適解を実数値として探索するために遺伝的アルゴリズムが使用される場合には、遺伝子型のコーディングとしては Gray code が用いられることが一般的である。それは以下の理由による。

例えば、(01111111)（整数 127）が最適解である場合に、(10000000)（整数 128）は値においては非常に最適解に近いが、通常 2 進数ではハミング距離が 7 であり、遺伝子型としては「距離が遠い」ため、適合度で十分に高い数値をえた (10000000) が集団中に拡散してしまう可能性がある。

Gray code ではおなじ実数値がそれぞれ (01000000)（整数 127）、(11000000)（整数 128）¹¹⁾ と表現されるからそのような問題は起こらない。

本稿で取り上げている問題は関数の最適化問題ではないが、本文中に述べたように、集団が協力へ収束するには「報恩の程度が高い ($p \approx 1.0$)」ことが重要になる。そうした個体の適合度が高いほど協力への収束が促されるため、今回の遺伝子型で言えば、「前半 10 ビットが示す実数値が 1,023 ($=2^{11}-1$) に近い」個体が多いほどそうした状況が実現しやすい。したがって、上位ビットに 1 が並んだ遺伝子型がよい得点を挙げていくことになる。

交叉や突然変異によって上位ビットのどれかが 1 から 0 へ変化したとして、通常 2 進数の表現では大きく変わる整数値が Gray code ではそれほど大きくは変わらない。

この事情から表 2 にあるように、Gray code でコーディングした場合にはどのケースにおいても通常 2 進数に比較すると協力を達成しやすくなった。

B 使用したツールについて

今回の分析では、遺伝的アルゴリズムにおける各世代ごとに個体分布を視覚化することが必要である。その際に、集団の平均適合度のグラフと対応した形で観察できることが望ましい。

考えられる手段としては遺伝的アルゴリズムによるシミュレーションプログラムの中に視覚化するための機能を実装することであるが、その場合には本論で取り上げたような「協力への収束」「一方の集団による他方の集団の搾取」といった事象が起きるかどうか分からない試行を繰り返しながら、そのような結果が出るのを待たねばならない。

そのためシミュレーションのメインとなる遺伝的アルゴリズムを実行するプログラムから、集団の平均適合度と各個体のもつ遺伝子型を外部ファイルへ書き出すようにし、そのファイルを使って視覚化するためのツールを別途開発した。

11) 違った文脈ではあるが、この例は De Jong (1992) で取り上げられている。

個体の遺伝子型は長さ 20 のバイナリ文字列である。集団は 100 個体で構成され、1,000 世代のループを回すことになるため、すべての遺伝子型をテキストファイルで保存しようとする、 $20 \times 100 \times 1,000 = 2 \times 10^6$ バイト必要であり、シミュレーション回数が例えば 1,000 回だとするとだいたい 2 GB の容量を消費することになる。

これは単一のテキストファイルとしては大きすぎるため、注目している事象（協力への収束と搾取）が起きた場合のみ、遺伝子型と集団の平均適合度を保存して、その事象が 100 回観測されたら終了するよう本体をプログラムすると共に、バイナリ文字列をいったん 16 進文字列に変換し、サイズを 4 分の 1 に圧縮している。そのファイルを分析用ツールで再びバイナリ文字列へデコードした上で実数値変換を行っている。

分析用ツールのスクリーンショットを図 7 に示す。

図中①が個体分布、②が集団平均適合度推移のグラフである。図では 2 集団ゲームの様子が描かれている。図中⑥はスライダーで、これを左右に操作することで世代 0 から世代 999 までの任意の世代を選択できる（図では 621 世代を選択。平均適合度グラフの垂直線）。世代の選択、実験番号の選択は④のテキストフィールドからでも可能である。⑤はログウィンドウで、パラメータや対象になっているファイル名などが確認できる。

任意の世代を選択すると、それに応じて①に個体分布が示されるため、②の平均適合度の推移をみながら特徴的な世代に垂直線をあわせて、その世代での分布を確認できるようになっている。スライダーを連続的に動かすと①にはアニメーションのように分布の変化が現れることになる。

そのうえで、特徴的な分布が見つければ③の左側ボタンで①に描かれている個体分布のグラフを外部ファイル（eps 形式）で出力することができる。③の右側ボタンは②の集団平均適合度推移グラフを出力する。

本稿で使用したグラフ類はすべてこのツールで作成した。

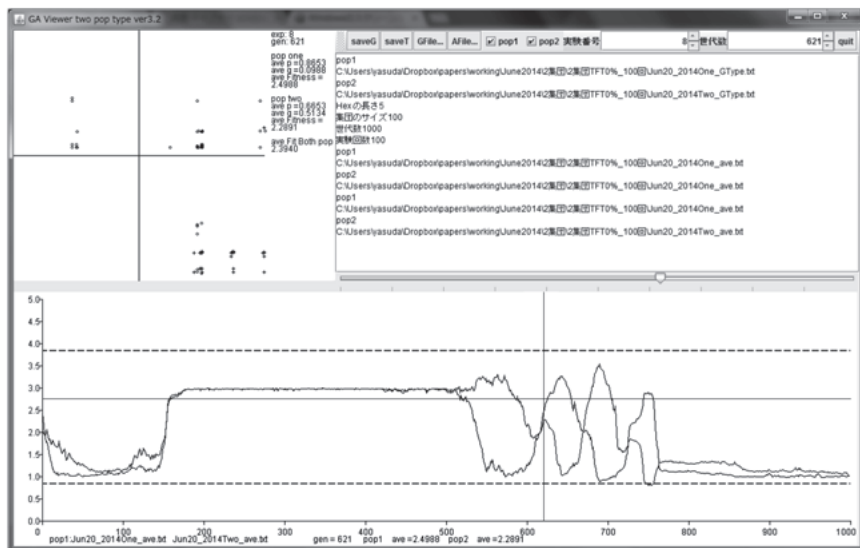


図7 シミュレーション結果分析用ツール

参 考 文 献

- Axelrod, Robert (1984) *The Evolution of Cooperation*, New York : Basic Books.
- (1997) *The Complexity of Cooperation* : Princeton University Press.
- De Jong, Kenneth A (1992) “Genetic Algorithms are NOT Function Optimizers.,” in *FOGA*, pp. 5-17.
- Nowak, Martin and Karl Sigmund (1990) “The Evolution of Stochastic Strategies in the Prisoner’s Dilemma,” *Acta Applicandae Mathematicae*, Vol. 20, pp. 247-265.
- (1992) “Tit for tat in heterogeneous populations,” *Nature*, Vol. 355, pp. 250-253, January.
- 安田俊一 (2001) 「RPDにおける戦略の進化—GAによる囚人のジレンマ実験(2)—」, 『松山大学論集』, 第13巻, 第3号, 27-43頁.
- (2005) 「どのような集団が協力を生み出すか? —GAによる囚人のジレンマ—」, 『進化経済学論集』, 第9巻, 217-226頁.